Emulating AQM From End Hosts

Sumitha Bhandarkar A. L. Narasimha Reddy
Yueping Zhang and Dmitri Lyuginov
August 30, 2007

Proactive Congestion Avoidance

Motivation

- TCP behavior :
  - Additive increase until packet loss is observed
  - Reduce cwnd by half ***after*** a packet loss
- Problem :
  - Bottleneck link buffers fill up
- Result :
  - Self induced packet losses
  - Wasted resources for retransmission of lost packets

Proactive Congestion Avoidance

Background

- Well understood problem
- Explicit congestion notification by router
  - RED, REM, AVQ, BLUE, VCP etc. with ECN
- Explicit rate control by router
  - XCP, RCP, EMKC, JetMax etc.
- End-host based solutions
  - CARD, TRI-S, DUAL, VEGAS, CIM etc.

Proactive Congestion Avoidance

Background (cont.)

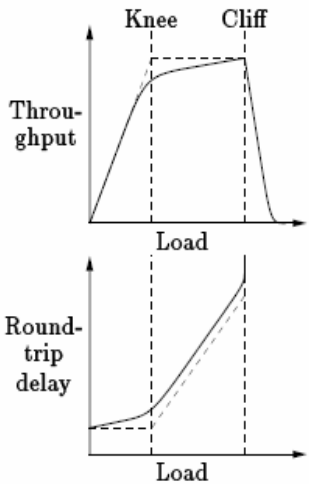
- Router based solutions
  - Easier to determine the onset of congestion
  - Difficult to deploy
- End-host based Solutions
  - Easier to deploy
  - Difficult to determine the onset of congestion
- We propose an end-host based solution that emulates router-based solution

Proactive Congestion Avoidance

Background (cont.)

End-host based prediction

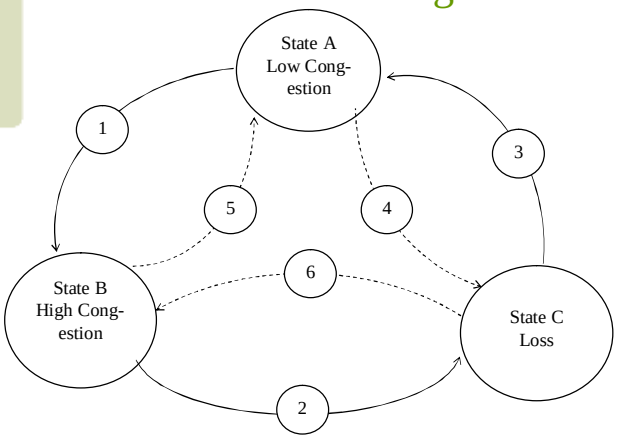
- Monitor throughput
  - Before link is full, throughput increases linearly with load
  - After link is full, throughput is constant at link capacity
- Monitor Delay
  - Before link is full, delay is low
  - After link is full, delay increases



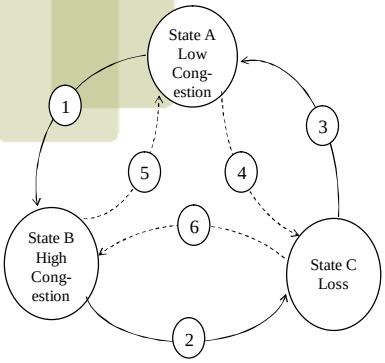
*Source: [CARD]

Proactive Congestion Avoidance

State Transition Diagram



Proactive Congestion Avoidance



Prediction Efficiency

$$\frac{(\text{“2” transitions})}{(\text{“2” transitions} + \text{“5” transitions})}$$

False Positives

$$\frac{(\text{“5” transitions})}{(\text{“2” transitions} + \text{“5” transitions})}$$

False Negatives

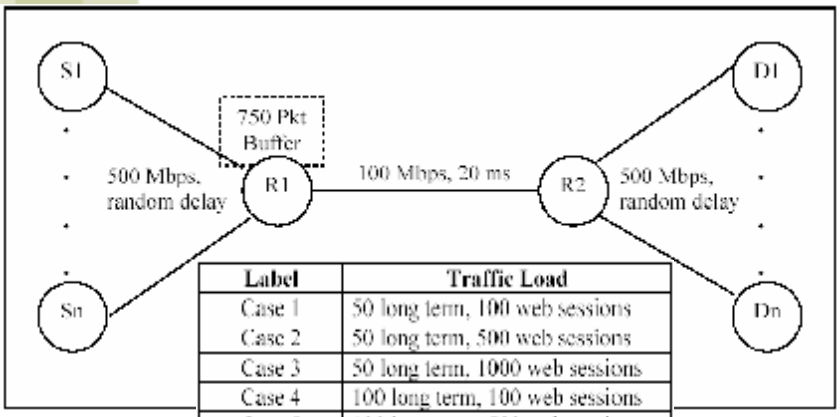
$$\frac{(\text{“4” transitions})}{(\text{“2” transitions} + \text{“4” transitions})}$$

Proactive Congestion Avoidance

- Measurement based studies claim…
  - End-host congestion avoidance not possible
  - Low correlation between RTT increase and loss
    - But loss measured at flow level
    - Absence of loss does *NOT* indicate absence of network congestion

Proactive Congestion Avoidance

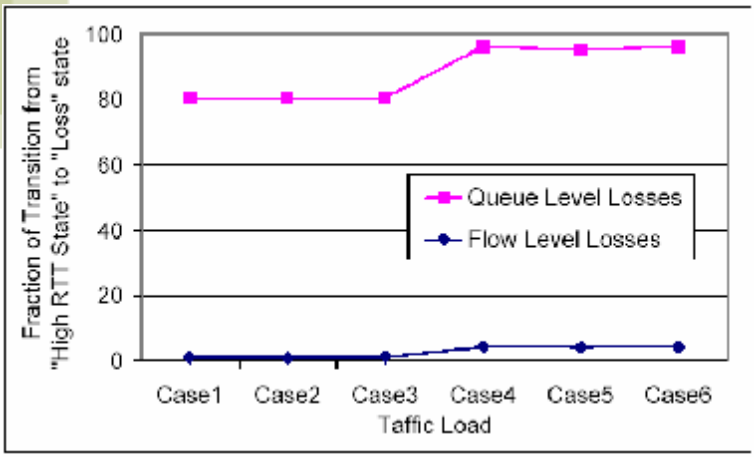
Correlation between RTT and Loss



Label	Traffic Load
Case 1	50 long term, 100 web sessions
Case 2	50 long term, 500 web sessions
Case 3	50 long term, 1000 web sessions
Case 4	100 long term, 100 web sessions
Case 5	100 long term, 500 web sessions
Case 6	100 long term, 1000 web sessions

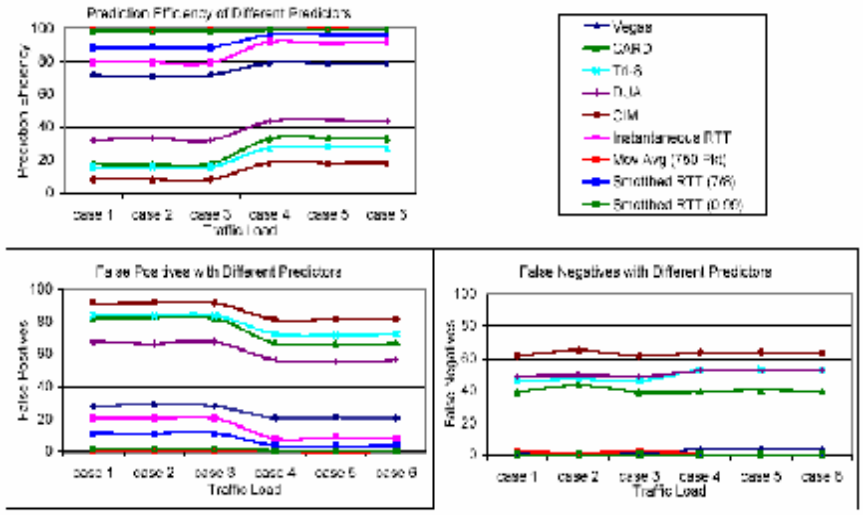
Proactive Congestion Avoidance

Correlation between RTT and Loss (Cont.)



Proactive Congestion Avoidance

Prediction Efficiency, False Positives & Negatives



Proactive Congestion Avoidance

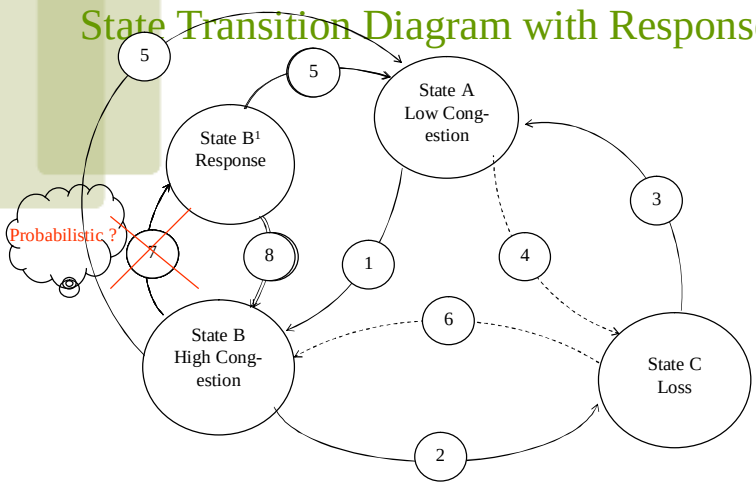
- Measurement based studies claim…
  - End-host congestion avoidance not efficient
  - Responding to uncertain signal can cause more harm than good
    - Assume response is multiplicative decrease with factor 0.5
    - Alternate response can be designed to provide robustness to uncertainties

Proactive Congestion Avoidance

- Designing the Response
  - False positives cannot be entirely eliminated
  - Response should be chosen such that impact of false positives can be reduced
    - When to respond ?
    - How to respond ?
    - How much response ?

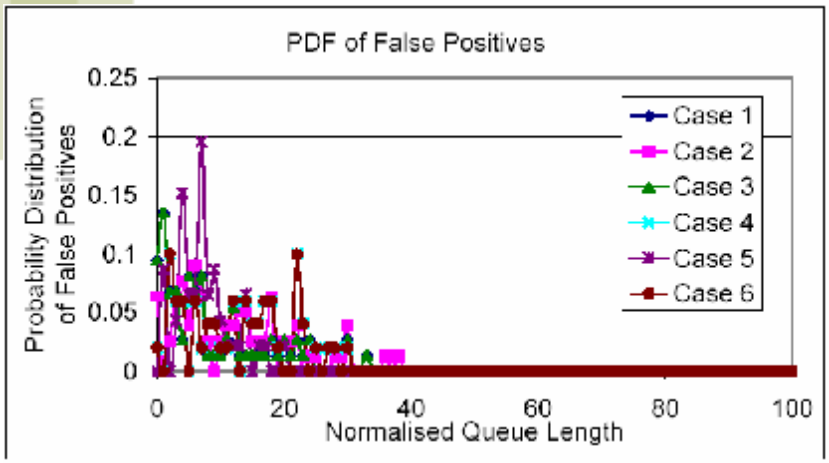
Proactive Congestion Avoidance

State Transition Diagram with Response



Proactive Congestion Avoidance

Designing the Probabilistic Response

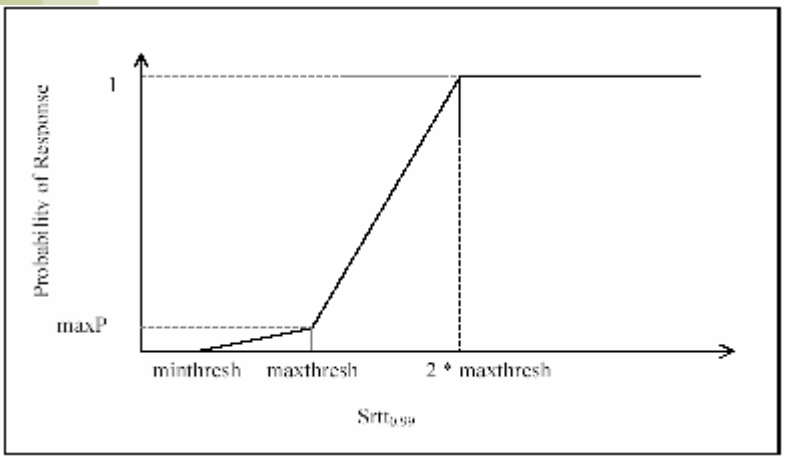


Proactive Congestion Avoidance

- Designing the Probabilistic Response
  - False positives decrease as queue length increases
  - Make probabilistic response a function of queue length
    - At lower queue length, probability of response is low
    - Probability of response increases as queue length increases
  - Conceptually, similar to RED/ECN
    - Emulate the RED probability curve
    - Use smoothed RTT for tracking queue length
    - Possible to emulate other AQM mechanisms also

Proactive Congestion Avoidance

Probabilistic Response Curve for PERT



Proactive Congestion Avoidance

- Probabilistic Early Response TCP (PERT)
  - Determine the appropriate prediction signal
    - improve the reliability of the prediction
  - Determine the appropriate response function
    - uncertainties in prediction are unavoidable
    - offset this uncertainty by making the response probabilistic
    - different AQM schemes can be emulated in the response - we choose RED/ECN

Proactive Congestion Avoidance

- Prediction signal used in PERT
  - RTT sample collected for every packet
  - Timestamp option used with high clock resolution
  - EWMA smoothing with weight 0.99 for history for eliminating noise
    - High prediction efficiency
    - Low false positives
    - Low false negatives

Proactive Congestion Avoidance

- Response function used in PERT
  - Probabilistic - emulates RED/ECN
  - Fixed values used for parameters
    - minthresh_ = 5ms
    - maxthresh_ = 10ms
    - maxP_ = 0.05
  - Adaptive values possible (similar to adaptive RED)
  - At most one response per RTT

Proactive Congestion Avoidance

- Response function used in PERT (cont.)
  - Multiplicative decrease with fixed window reduction factor of 0.35
    - Buffer size B related to window reduction factor f as

$$B > \frac{f}{1-f} * BDP$$

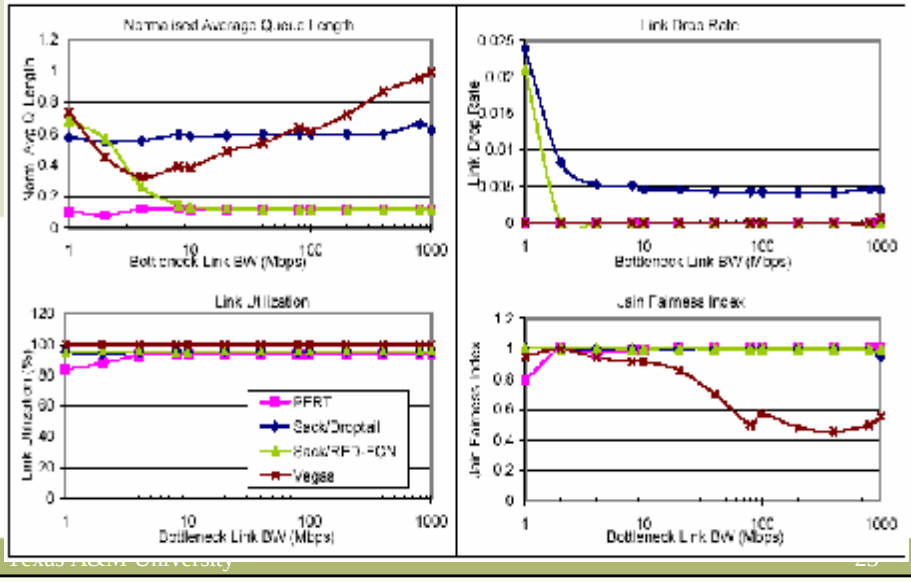
    - Generally buffer size is set to one BDP
    - If buffers do not exceed 0.5 BDP, $f = 0.35$ sufficient
  - If packet loss occurs, response similar to TCP
    - Window reduction by 0.5, fast retransmit/recovery.

Proactive Congestion Avoidance

- Experimental Evaluation
  - Extensive evaluation based on ns-2 simulations
  - Single bottleneck link
    - Bandwidth varied in the range [1Mbps, 1Gbps]
    - RTT varied in the range [10ms, 1s]
    - # long term flows varied in the range [1,1000]
    - # web sessions varied in the range [10,1000]
  - Multiple Bottleneck Link
  - Flows with different RTTs
  - Impact of sudden changes in traffic load

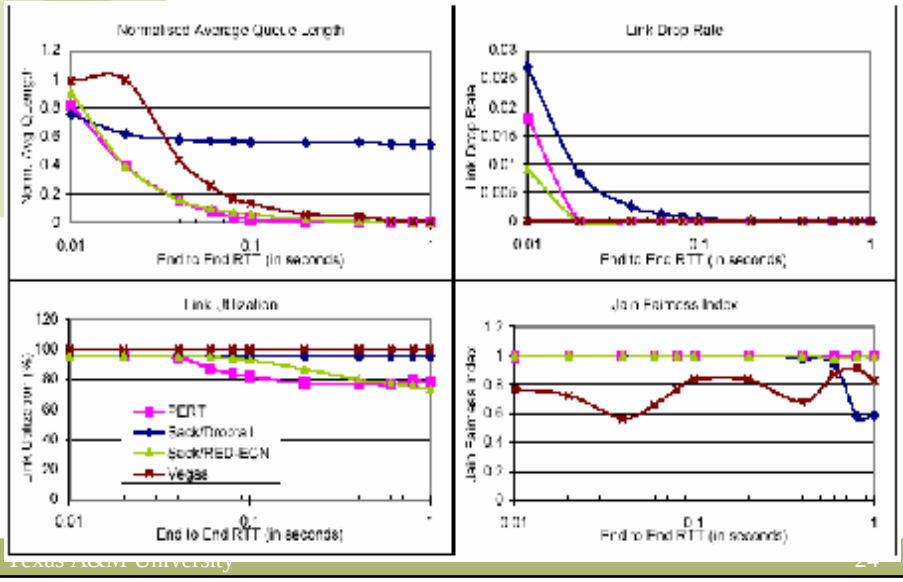
PERT : Experimental Evaluation

Varying the Bottleneck Link Bandwidth



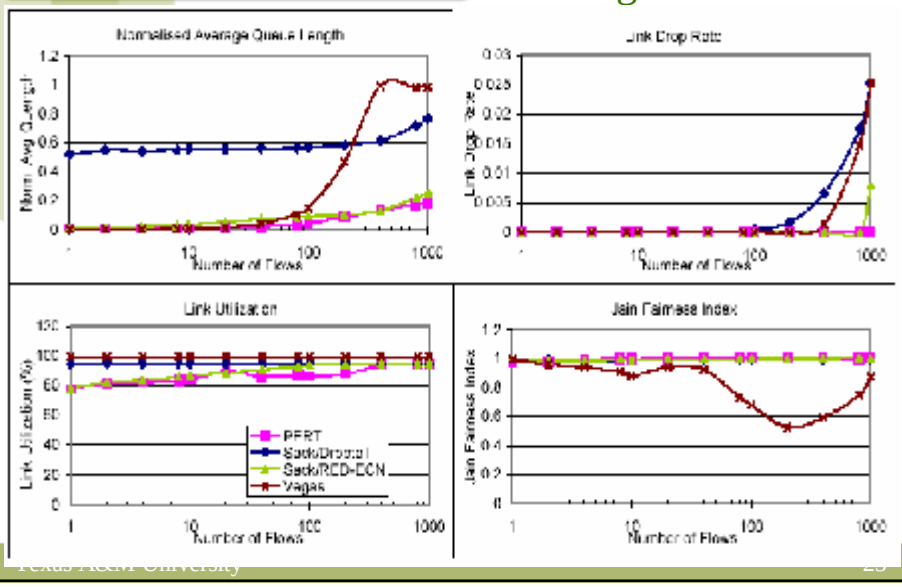
PERT : Experimental Evaluation

Varying the RTT



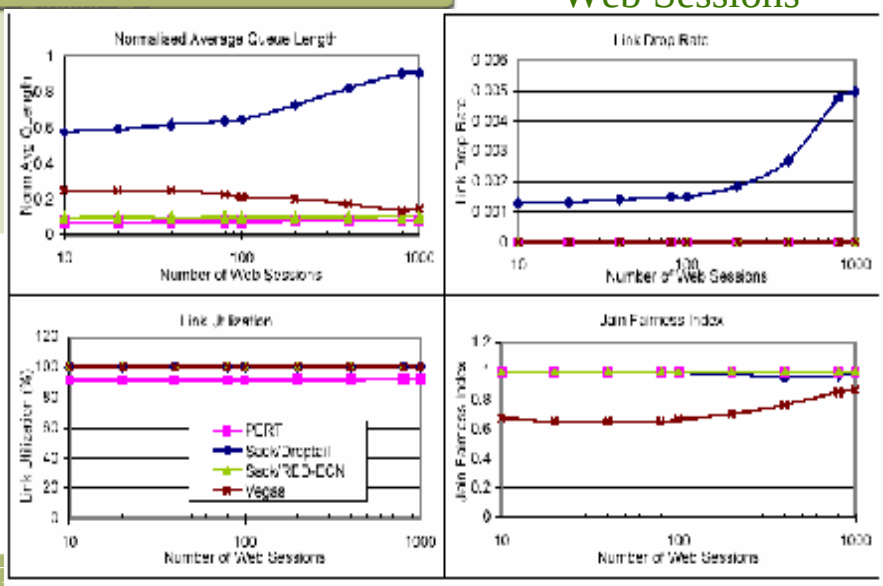
PERT : Experimental Evaluation

Varying the # Long-term Flows



PERT : Experimental Evaluation

Varying the # Web Sessions



PERT : Experimental Evaluation

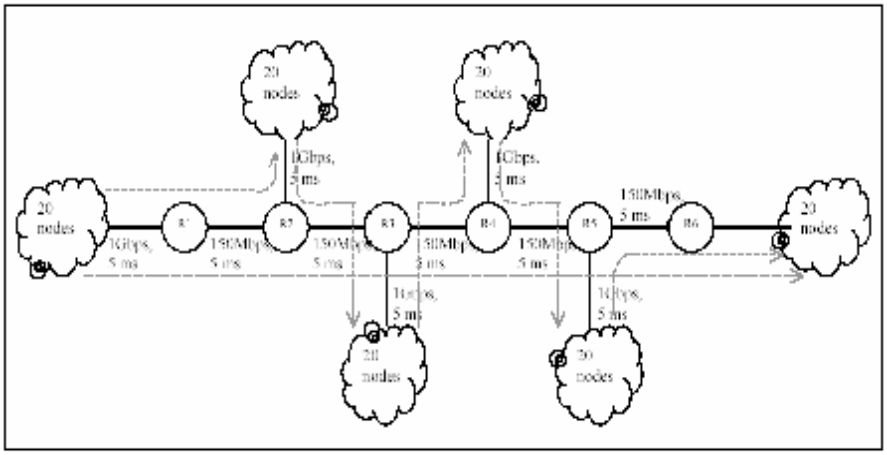
RTT Unfairness

Bottleneck link shared by 10 flows
with RTTs 12ms, 24ms, 36ms… 120ms

	PERT	Sack / Droptail	Sack / RED-ECN	Vegas
Link Utilization	93.81	93.77	93.90	99.99
Norm Avg Qlength	0.28	0.42	0.41	0.07
Link Droprate	3.98E-06	7.18E-04	4.95E-04	0
Jain Fairness Index	0.86	0.44	0.51	0.98

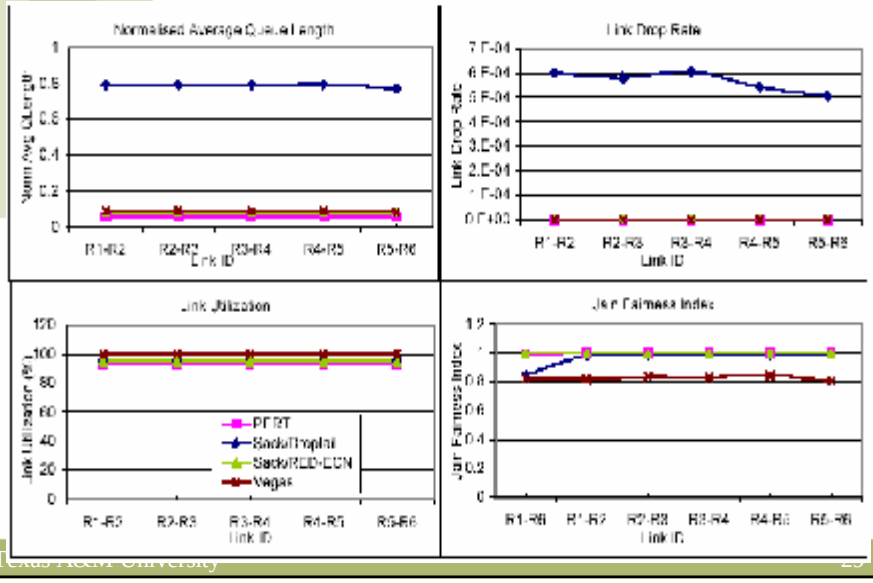
PERT : Experimental Evaluation

Multiple Bottleneck Links



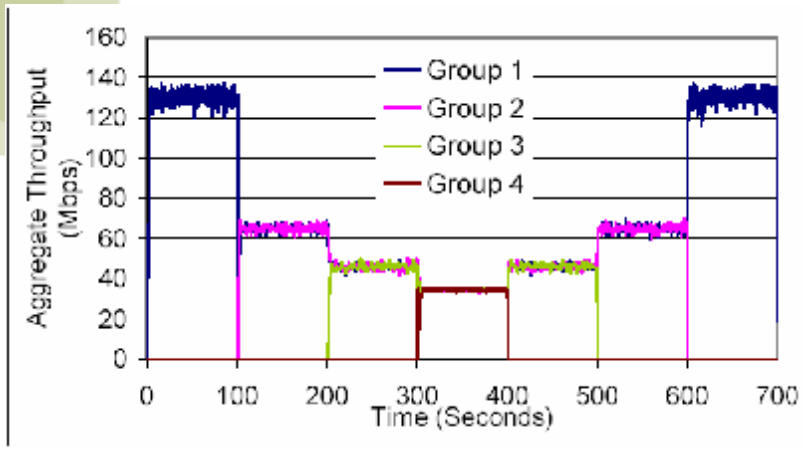
PERT : Experimental Evaluation

Multiple Bottleneck Links



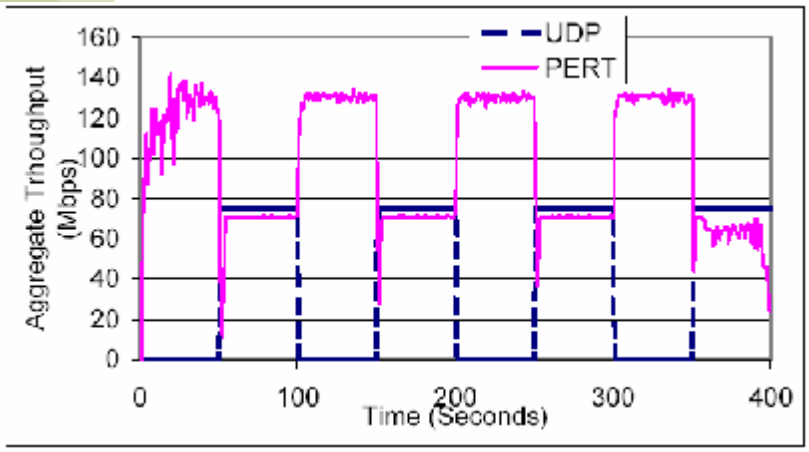
PERT : Experimental Evaluation

Dynamic Changes in Traffic



PERT : Experimental Evaluation

Sudden Changes in Non-responsive Traffic



Modeling – Window Dynamics

- Modeling of PERT is composed of three parts, window dynamics, RED emulation, and queuing behavior
- Consider a single-link network with propagation delay T_p, queuing delay $T_q(t)$, and round-trip delay $R(t) = T_p + T_q(t - R(t))$
- Denoting by $p(t)$ loss probability, window dynamics $W(t)$ can be described by:

$$\dot{W}(t) = \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t-R(t))}p(t)$$

Modeling – RED Emulation

- Drop rate p(t) is computed by

$$p(t) = \frac{T_q(t) - T_{min}}{T_{max} - T_{min}} p_{max}$$

- To obtain $T_q(t)$, the router needs to estimate the RTT R(t), which is given by:

$$R(t) = \alpha R(t-1) + (1-\alpha)\hat{R}(t)$$

constant

instantaneous RTT

- According to Hollot's SIGCOMM00 paper, the last equation can be approximated by:

$$\dot{R}(t) = \frac{\ln \alpha}{\delta}(R(t) - \hat{R}(t))$$

Modeling – Queuing Dynamics

- Queuing dynamics q(t) can be represented by the following differential equation:

$$\dot{q}(t) = \frac{W(t)}{R(t)} N(t) - C$$

number of flows

link capacity

- Since $T_q(t)=q(t-R(t))/C$, we can rewrite the last equation in terms of queuing delay $T_q(t)$:

$$\dot{T}_q(t) = \frac{W(t-R(t))N(t-R(t))}{R(t-R(t))C} - 1$$

- Then, we obtain the complete system model of PERT/RED

Modeling – Complete Model

- Assuming N and R are constant in the steady state and denoting W(t -R(t) by $W_R(t)$, the system model becomes:

$$\begin{cases} f(W, W_R, T_q, p) & = \dfrac{1}{R} - \dfrac{W(t)W_R(t)}{2R}p(t) \\ g(W, T_q) & = \dfrac{N}{RC}W(t) - 1 \end{cases} \quad (*)$$

- We have the following equilibrium points:

$$W^* = \frac{RC}{N} \text{ and } p^* = \frac{2N^2}{R^2C^2}$$

- We next study local stability of (*)

Stability Condition

- <u>Theorem 1</u>: Let L_{PERT} and K be defined as:

$$L_{PERT} = \frac{p_{max}}{T_{max} - T_{min}}, \quad K = \frac{\ln \alpha}{\delta},$$

and assume bounds R^+ and N^- satisfy the following condition:

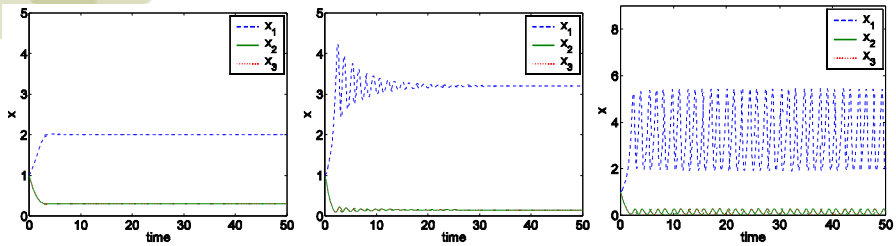
$$\frac{L_{PERT}R^{+3}C^2}{(2N^-)^2} \leq \sqrt{\frac{w_g^2}{K^2} + 1},$$

where $w_g = 0.1 \min\left(\frac{2N^-}{R^{+2}C}, \frac{1}{R^+}\right).$

Then, PERT modeled by (*) is locally stable for all $N \geq N^-$ and $R^* \leq R^+$

Simulations

- Set link capacity $C=100$ pkt/s, $N=N^{-}=5$, $\delta=0.1$ ms, $p_{max}=0.1$, $T_{max}=100$ ms, $T_{min}=50$ ms, and $\alpha=0.99$. Stability boundary is $R=171$ ms



R = 100 ms R = 160 ms R = 171 ms

- Stability condition in Theorem 1 is tight

Summary of Analysis

- PERT-RED has larger stability region than RED
  - Queuing delay versus queue lengths
  - Rate of sampling queue at C/n versus C
- PERT-RED can be unstable at higher delays
- PERT-PI has similar advantages over PERT-RED as PI over RED

Emulating PI

- The discrete PERT/PI is:

$$p(t) = \beta(T_q(t) - T_q^*) - \gamma(T_q(t-1) - T_q^*) + p(t-1)$$

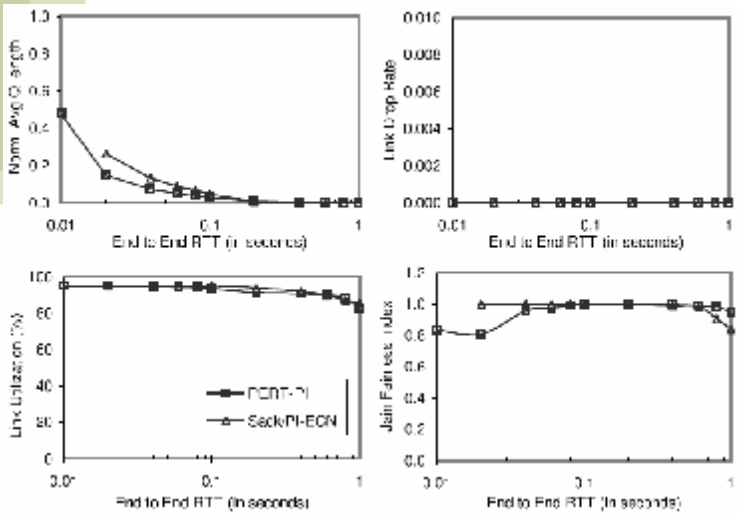
where T_q^* is the target queuing delay, m is a constant, $\gamma = K/m + K\delta/2$, and $\beta = K/m - K\delta/2$

- <u>Theorem 2</u>: Assuming $W^* >> 2$, PERT/PI with

$$m = \frac{2N^-}{R^{+2}C} \quad \text{and} \quad K = m\left|\frac{jR^*m + 1}{\frac{R^{+3}C^2}{(2N^-)^2}}\right|$$

is locally stable for all $N \geq N^-$ and $R^* \leq R^+$

PERT/PI vs. AQM/PI



Proactive Congestion Avoidance

- In summary..
  - End-host based congestion prediction more accurate than previously characterized
  - Further improvement in congestion prediction signal possible
  - Not all uncertainty can be eliminated - need “smart” response mechanism
  - One possible choice is probabilistic response similar to RED/ECN
  - Benefits similar to RED/ECN, some open issues to be addressed